

IN THE CLAIMS

Following are the claims as currently pending for consideration.

1-13. (Canceled)

14. (Original) A multiprocessor system comprising:

means for receiving one or more semaphore modification requests from one or more requesting devices;

means for identifying an ownership state of a semaphore corresponding to the one or more semaphore modification requests;

means for arbitrating to identify a first modification request of the one or more semaphore modification requests, the first modification request from a first requesting device;

means for granting the first modification request if the identified ownership state corresponds to the first requesting device; and

means for granting the first modification request if the identified ownership state corresponds to no owner.

15. (Original) The multiprocessor system recited in Claim 14 further comprising:

means for receiving a semaphore read requests from one of the one or more requesting devices;

means for transmitting the identified ownership state in response to the semaphore

read request; and

means for allowing the first requesting device to access a shared resource.

16. (Original) The multiprocessor system recited in Claim 14 wherein the one or more requesting devices are fabricated on a single die.

17. (Previously Presented) A multiprocessor comprising:

a logical plurality of processors;

a resource scheduling device coupled to one or more of the logical plurality of processors to provide access to a set of resources;

a shared resource of the set of resources having a semaphore; and

a semaphore checker coupled to the resource scheduling device and to the semaphore to:

receive one or more semaphore modification requests from the one or more of the logical plurality of processors,

identify an ownership state of the semaphore,

arbitrate the one or more semaphore modification requests and identify a first modification request from a first requesting processor of the one or more of the logical plurality of processors,

allow the first modification request to succeed if the identified ownership state corresponds to the first requesting processor and

allow the first modification request to succeed if the identified ownership state corresponds to no ownership.

18. (Original) The multiprocessor recited in Claim 17 wherein the semaphore checker is further to:
- decline a second modification request of the one or more semaphore modification requests.
19. (Original) The multiprocessor recited in Claim 17 wherein the semaphore checker is further to:
- receive a semaphore read requests from one of the one or more of the logical plurality of processors;
 - transmit the identified ownership state in response to the semaphore read request;
 - and
 - allow the first requesting processor to access a shared resource.
20. (Original) The multiprocessor recited in Claim 17 wherein each of the one or more semaphore modification requests received identify a corresponding requesting processor of the one or more of the logical plurality of processors.
21. (Original) The multiprocessor recited in Claim 17 wherein the multiprocessor is fabricated on a single die.
22. (Original) The multiprocessor recited in Claim 17 wherein arbitration is resolved on a round-robin basis.

23. (Original) The multiprocessor recited in Claim 17 wherein arbitration is resolved on a priority basis.

24. (Original) An apparatus comprising:

- a register to access a shared resource of a set of resources;
- a semaphore corresponding to the shared resource; and
- a semaphore checker coupled to the semaphore to allow access to the shared resource through the register.

25. (Previously Presented) The apparatus of Claim 24 wherein the semaphore checker is further to:

- receive one or more semaphore modification requests from one or more of a logical plurality of processing devices,
- identify an ownership state of the semaphore,
- arbitrate the one or more semaphore modification requests and identify a first modification request from a first requesting device of the one or more of the logical plurality of processing devices to succeed if the identified ownership state corresponds to the first requesting device; and
- allow the first modification request to succeed if the identified ownership state corresponds to no ownership.

26. (Original) The apparatus of Claim 25 wherein the semaphore checker is further to:
decline a second modification request of the one or more semaphore modification requests.
27. (Original) The apparatus of Claim 25 wherein the semaphore checker is further to:
receive a semaphore read requests from one of the one or more of the logical plurality of processors;
transmit the identified ownership state in response to the semaphore read request;
and
allow the first requesting processor to access a shared resource.
28. (Currently Amended) ~~The apparatus of Claim 25~~ An apparatus comprising:
a register to access a shared resource of a set of resources;
a semaphore corresponding to the shared resource; and
a semaphore checker coupled to the semaphore to allow access to the shared resource through the register, the semaphore checker further to:
receive one or more semaphore modification requests from one or more of
a logical plurality of processing devices, wherein each of the one or more semaphore modification requests received identify a corresponding requesting device of the one or more of the logical plurality of processing devices,
identify an ownership state of the semaphore,
arbitrate the one or more semaphore modification requests and identify a
first modification request from a first requesting device of the one or more of the

logical plurality of processing devices to succeed if the identified ownership state corresponds to the first requesting device; and

allow the first modification request to succeed if the identified ownership state corresponds to no ownership.

29. (Original) The apparatus of Claim 25 wherein the logical plurality of processing devices are integrated on a single die.

30. (Original) The apparatus of Claim 25 wherein arbitration is resolved on a round-robin basis.